Holmes: Automated Fact Check with Large Language Models

Haoran Ou*, Gelei Deng*, Xingshuo Han*, Jie zhang[†], Xinlei He[‡], Han Qiu[§], Shangwei Guo[¶] and Tianwei Zhang*

*Nanyang Technological University

[†]CFAR and IHPC, A*STAR, Singapore

[‡]The Hong Kong University of Science and Technology (Guangzhou)

[§]Tsinghua University

¶Chongqing University

Abstract—The increased Internet connectivity inevitably leads to the rapid proliferation of disinformation on digital platforms, which poses significant threats to societal trust, individual decision-making, and even national security. Disinformation is becoming more sophisticated, transitioning from the single-text modality to complex multimodality involving images and text. This evolution makes existing solutions struggle in multimodal disinformation detection: Conventional deep learning methods exhibit weak representational capacity and perform inadequately in learning the complex features in multimodal disinformation.

Inspired by the recent advances in AI technology, this study investigates the opportunities of leveraging Large Language Models (LLMs) for automated disinformation detection. We first conduct an empirical study to assess the capability of state-of-the-art LLMs in verifying multi-modal disinformation. We find that (1) LLMs fall short of identifying the truthfulness of disinformation just based on the target claim; (2) by supplementing LLMs related evidence, their detection capability is significantly enhanced; (3) unfortunately, LLMs are incapable of searching for accurate and relevant evidence autonomously.

To better solve this problem, we propose Holmes, an end-toend framework for automated fact-checking and disinformation detection. The core of Holmes is a novel evidence retrieval methodology that guides LLMs in collecting high-quality evidence for better disinformation detection. This approach consists of two innovations: (1) We harness the summarization capability of LLMs to extract the main content from the open-source information following some specific rules. (2) We propose a new algorithm and metrics to evaluate the quality of extracted evidence. The evidence generated from these two steps can assist LLMs in verifying the claim and generating the verdict and justification accurately. We perform extensive experiments to validate the effectiveness of Holmes in verifying multimodal disinformation. It achieves an accuracy of 88.3% in two opensource datasets and 90.2% in a real-time disinformation verification task. Particularly, our improved evidence retrieval technique can boost the accuracy of fact-checking by 30.8% compared to existing methods.

I. Introduction

With the rise of social media, it becomes increasingly easy for individuals to express their opinions freely, leading to a surge in online content. A significant portion of this content constitutes disinformation, which is misleading, deceptive, or malicious. It can spread rapidly worldwide, presenting considerable risks to public safety, social trust, and national security. For example, during COVID-19, a lot of disinformation was

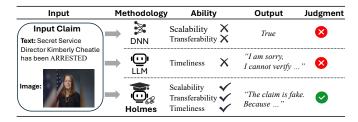


Fig. 1: DNN-based methods underperform in verifying the claim due to the limited model size and bad transferability. LLMs fail to verify the disinformation that is published later than their knowledge cutoff date. Holmes overcomes these weaknesses and achieves the best performance.

propagated online [1], advocating vaccine avoidance, mask refusal, and use of medications with insignificant scientific data. This ultimately led to public health crises and overwhelmed healthcare systems [2].

Therefore, it is imperative to detect disinformation promptly, halt its spread, and trace its origins. Different countries and regions make great efforts to combat disinformation. The European Union (EU) established the European Digital Media Observatory (EDMO) [3], which works in collaboration with fact-checking organizations to tackle disinformation. The Poynter Institute Florida in the United States launched the International Fact-Checking Network (IFCN) [4] in 2015 to provide a fact-checking service to the public, making them better informed.

Existing misinformation detection solutions primarily focus on textual content, and utilize deep learning techniques such as Recurrent Neural Networks (RNN) [5], Convolutional Neural Networks (CNN) [6], and attention mechanisms [7]. However, today's disinformation has become more sophisticated, incorporating multiple modalities such as images. This evolution allows for the presentation of misleading content in more compelling formats, significantly enhancing its reach and impact across various platforms. To address this, researchers have developed new multimodal disinformation detection methods. For example, some studies [8], [9] aim to align textual and visual features to assess cross-modal consistency and verify

the authenticity of information. Others [10], [11] introduce supplementary evidence related to the target information to enhance the detection accuracy and generalization. Nonetheless, these traditional deep learning-based approaches are often constrained by the model size and training dataset, resulting in poor transferability when confronting new forms of disinformation, as shown in Figure 1.

The recent advances of AI technology, particularly Large Language Models (LLMs), bring new opportunities for disinformation detection. LLMs have demonstrated impressive performance in a variety of tasks, including reasoning [12], [13], text generation [14], and multimodality processing [15]. Researchers have also made some preliminary attempts to apply LLMs for disinformation detection, following two manners: designing specific instructions to prompt LLMs for misinformation detection [16] and integrating LLMs into existing frameworks to improve the overall detection accuracy [17]. Unfortunately, these solutions still focus on the text modality. How to detect the misinformation with the text and image modalities with LLMs is still underexplored.

Our contribution. To address these limitations, this paper presents an in-depth investigation towards LLM-assisted automated disinformation detection (Figure 1). Our contributions are twofold: First, we conduct comprehensive evaluations to assess the potential and capability of LLMs in verifying multimodal disinformation. We establish a dataset by collecting samples from reliable and authoritative fact-checking sources. We then implement two types of claim verification strategies [11]: (1) Unsupervised disinformation verification: There is no human involvement in the verification process, as the LLMs independently verify the disinformation. (2) Supervised disinformation verification: LLMs verify disinformation with human assistance. Human involvement occurs either by guiding the LLMs through a structured fact-checking pipeline or by providing human-written evidence as external knowledge to support the verification process. For each strategy, we evaluate the final results from the LLM, including verdict and justification, which delineate the logical process of disinformation verification.

We derive several intriguing observations from the above assessment: (1) Despite excellent reasoning capabilities, LLMs struggle to verify the truthfulness of disinformation solely using their inherent knowledge base, even when enhanced with the Chain-of-Thought prompting technique [12]. (2) Despite being guided by a fact-checking pipeline, LLMs cannot reliably verify claims because they lack access to necessary information, leaving the results they generate are often invalid. This limitation usually results in the generation of inaccurate or fabricated evidence, known as hallucination [18]. (3) When provided with sufficient evidence, LLMs demonstrate improved accuracy in fact checking.

Based on these observations, the *second* contribution is the design and development of an end-to-end framework, Holmes, for real-time verification of multimodal disinformation. The core component of Holmes is a novel evidence retrieval methodology, which can obtain high-quality and rel-

evant evidence for LLMs to refer to. Specifically, Holmes first decomposes a multimodal claim into several single-modal subclaims. It then utilizes third-party tools to search the Internet for relevant information corresponding to these sub-claims. Subsequently, it employs an LLM to summarize the main content of the identified web pages following standardized rules and construct the evidence. If a web page is inaccessible, Holmes uses the snippet (namely, the brief textual summary that search engines return alongside search results) as a proxy of the original full page content. A new algorithm is introduced to evaluate the quality of the evidence and identify the best one. With such compiled evidence, the LLM can confidently determine the truthfulness of the claim. It can deliver a verdict accompanied by a step-by-step justification, outlining its reasoning process in the required format.

Holmes offers three notable advantages. First, unlike traditional deep learning-based methods, Holmes relies solely on off-the-shelf LLMs without requiring data labeling or model training, significantly reducing development costs. Second, it addresses the limitations of LLMs while fully harnessing their reasoning and summarization capabilities, enabling the rapid verification of real-time information—key to countering the swift spread of disinformation. Third, Holmes provides detailed justifications and explainability for its verdicts, enhancing the credibility of the verification results and facilitating user understanding.

We evaluate the effectiveness of Holmes through extensive experiments and rigorous analysis. The evaluation results demonstrate that Holmes achieves an accuracy of 88.3% over two open-source multimodal disinformation datasets [11], [19], and exhibits stronger generalization compared to four deep learning-based baseline methods [20], [21], [22], [23]. Additionally, we assess the real-time verification capability of Holmes using a self-constructed dataset. It achieves a 90.2% success rate in automatically retrieving the evidence and verifying claims, with a cost of 0.11 USD and 11.8 seconds per claim. Finally, we compare the efficiency of our evidence retrieval method with previous solutions [24], [20], validating its benefit of enhancing the detection accuracy.

II. BACKGROUND OF DISINFORMATION DETECTION

Disinformation broadly refers to fabricated or intentionally manipulated text, speech, visuals, conspiracy theories, or rumors [25]. Its dissemination online undermines public trust and leads to harmful societal outcomes in critical domains such as health, politics, and safety. As disinformation evolves, accurate detection becomes increasingly critical. Current detection methods primarily include direct disinformation detection and automated fact-checking.

A. Disinformation Detection

Disinformation detection methods directly identify false information, shifting recently from text-based approaches [26], [27], [28], [29] towards multimodal methods that integrate text and image information [8], [9], [30]. These multimodal approaches often use fusion networks and cross-modal alignment

techniques to address cross-modality ambiguity. For example, Wang et al. [31] introduced COOLANT, a model combining cross-modal contrastive learning and attention mechanisms, achieving state-of-the-art performance on Twitter and Weibo datasets. However, these methods face limitations such as poor generalization to diverse writing styles and limited explainability, as models usually provide predictions without reasoning or justification.

B. Automated Fact Check

To enhance interpretability and reliability, automated fact-checking (AFC) approaches verify claims by explicitly retrieving and evaluating related evidence. A typical AFC pipeline includes three primary stages [32]: claim detection and extraction, evidence retrieval, and verdict prediction with justification. Compared to direct detection methods that solely focus on claim content, AFC significantly improves detection accuracy by systematically leveraging external evidence [33].

A critical aspect of AFC is evidence retrieval, with existing methods generally divided into two categories based on evidence sources. The first category retrieves relevant evidence from structured knowledge bases or corpora, such as Wikipedia or fact-checking archives, by measuring semantic similarity between claims and corpus content [19], [17], [34]. For instance, Wu et al.[35] proposed an evidence-enhanced inference framework employing dual-level keyword retrieval to identify relevant articles and sentences. The second category leverages general-purpose search engines to retrieve supporting evidence directly from the Internet [24], [20]. Numerous deep learning-based AFC models have been developed, integrating multimodal representations of claims and evidence to verify credibility [10], [11], [19], [32], [33]. For example, Ma et al. [36] introduced a hierarchical attention network, which employs coherence-based and entailmentbased attention mechanisms to represent evidence coherently and assess its semantic alignment with claims. Other works further enhance AFC performance through extracting finergrained semantic features [36], [37], [38] or by improving the quality and reliability of retrieved evidence [10], [33].

Despite advancements, AFC approaches continue to face notable limitations. First, they often struggle with cross-domain generalization, encountering reduced accuracy when applied to novel topics, languages, or contexts. Second, deep learning-based AFC methods tend to be resource-intensive and require significant effort to construct reliably annotated datasets and to train robust models [39], [40]. Lastly, the retrieved evidence itself may be problematic. Evidence drawn from corpora can be limited or outdated, while evidence retrieved through web search might be incomplete, irrelevant, or unreliable, ultimately degrading verification effectiveness.

C. Disinformation Detection and AFC with LLMs

Recent development of LLMs, with their strong ability of text generation and reasoning, brings new opportunities for disinformation detection and fact check. Some works [41], [42] highlight the potential of LLMs like GPT-3.5 in fake news

detection, although they are still less accurate than human fact-checkers. Various template prompts and frameworks are further designed meticulously to improve LLMs' performance of disinformation verification. For example, Zhang et al. [16] introduced a few-shot prompting approach, which combines a hierarchical step-by-step (HiSS) method, to significantly enhance the accuracy of news verification, outperforming traditional supervised models. Pan et al. [17] proposed to decompose the complex claim into multiple subtasks, and verify them one by one through a well-designed workflow.

The above-mentioned works have not fully unleashed the power of LLMs in disinformation detection. Besides, they primarily focus on textual disinformation detection, yet the detection of multimodal disinformation with LLMs remains unexplored. In this paper, we aim to conduct a thorough study to bridge such gaps.

III. THREAT MODEL

In this work, we consider a threat model where an adversary deliberately fabricates and disseminates disinformation in textimage combinations. The adversary is assumed to have the ability to generate nonfactual content by manipulating existing media or composing misleading combinations of real elements. For instance, the adversary may pair authentic photographs with misleading or fabricated captions to create deceptive narratives. The victims are the general public, who are exposed to and potentially influenced by such content. The adversary's goal is to mislead public opinion, sow confusion, or promote specific agendas that deviate from factual information.

From the defender's perspective, we do not assume any prior knowledge about specific disinformation instances. Instead, we assume the defender develops fact-checking algorithms based on large language models (LLMs), with access to publicly available web resources. The primary objective of the defender is to detect and verify the veracity of multimodal claims in a real-time and automated manner.

Therefore, this work investigates the potential of LLMs in detecting multimodal disinformation and aims to design an automated fact-checking system grounded in LLMs. To achieve this, we first conduct an empirical study to assess the native capabilities of LLMs in detecting multimodal disinformation. Based on the findings, we propose a novel framework that automatically retrieves evidence, verifies claims, and produces interpretable justifications for its decisions.

IV. EMPIRICAL STUDY

To explore LLMs' capabilities in multimodal disinformation detection, we conduct a series of experiments to answer the following two research questions:

- (RQ1) Can LLMs directly verify the truthiness of given claims?
- (RQ2) Can LLMs verify the truthiness of given claims guided by the fact-checking pipeline?
- (**RQ3**) Can additional evidence improve the performance of LLMs in disinformation verification?

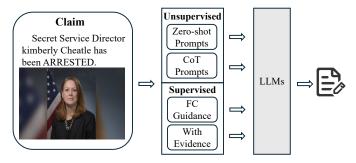


Fig. 2: Four strategies with LLMs for disinformation verification.

Figure 2 shows our evaluation strategies. We implement four distinct strategies to evaluate the performance of LLMs in verifying disinformation. These strategies are classified into unsupervised and supervised settings, depending on whether human guidance or human-provided evidence is involved in the verification process. We detail the evaluation details below.

A. Experimental Setup

1) Dataset Construction: We build a dataset containing true and false information for our evaluation. We follow two basic rules during the construction process: (1) Trustworthiness: the dataset only contains verified news and disinformation as valid baselines. (2) Timeliness: the release date of the samples in the dataset is relatively recent and not be included in the training set of the selected LLMs, so that we rule out potential biases arising from prior exposure to the disinformation.

To meet the trustworthiness requirement, we gather samples verified by reputable fact-checking agents. When disinformation appears on the Internet and raises significant public concern, authoritative entities, such as government agencies, respond promptly to address these issues and maintain social stability. We consider samples that have undergone such factchecking as validated and incorporate them into our dataset. Reuters¹, a popular news agency, serves as our primary data source due to its global reputation for impartiality, accuracy, and integrity in journalism. It has a "news" column that categorizes news into different sections according to their content. To ensure balance in the dataset, we collect an equal number of news sampled from various categories. We use the news caption as the claim to be verified and label it as true. Additionally, Reuters has a "fact-checking" column that examines the narratives and disinformation circulating on social media (e.g., Twitter, Facebook). In each article, Reuters journalists introduce basic information about the disinformation, such as the claim content, its source, and the source URL. We collect these original disinformation samples, including text and images, as negative examples. At the end of each fact-checking article, the authors provide a verdict, which includes labels such as false, satire, misleading, and others. We map all such labels to the false category. For the timeliness requirement, we review the release dates of the samples and filter out those that were published before the cutoff date of the LLM training sets². This ensures that the selected samples are not in the knowledge base of the LLMs.

Following those principles, we manually collect 113 samples from Reuters and each sample consists of a claim and label. The composition and property of the dataset are shown in Table I. There are 6 original verdicts of the disinformation and 1 verdict of news. To balance the distribution of the disinformation and true news in the dataset [43], we select 73 positive examples and 40 negative ones. All of the samples are collected from articles released between February 2024 and August 2024 which are later than the knowledge cutoff dates of GPT-40 and Gemini-1.5-flash.

For retrieval-based verification, we extend the dataset by supplementing additional evidence for each claim. These evidences are collected from the original fact-check columns validated by Reuters. In particular, Reuters journalists collect relevant materials related to the disinformation from authoritative and trustworthy online sources and then summarize the main points of the articles into abstracts, which serve as the final verdict for the claim verification. We collect such evidence and break it into several paragraphs according to its source and content. Meanwhile, we also collect the justification which could support the verdict for further analysis of the experimental results.

2) Evaluation Strategy: We select two state-of-the-art LLMs that support the image modality: GPT-40 [44] and Google Gemini-1.5-flash [45]. We aim to assess whether the two models could (1) verify the truthfulness of claims; (2) summarize logical reasons that support their verification.

For RQ1, the evaluation pipeline is shown in the top 2 rows of Figure 2. We deploy two prompting techniques for: zero-shot prompting [13] and Chain-of-Thoughts (CoT) [12]. These two approaches have been widely applied to various tasks, such as question answering, text understanding, and mathematical reasoning [12], [13]. The evaluation process begins by preparing the multimodal claim, which includes text and, if applicable, an associated image. For zero-shot prompting, the LLM is instructed to directly assess the claim's truthfulness using a standardized prompt: "Please verify the following claim. If you can verify the truthfulness of the claim, answer with 'yes' and explain why it is true or false. If you cannot verify it, answer with 'no' and provide the reason." The LLM's response is subsequently normalized for consistent analysis. For CoT, the LLM is guided to generate several logical steps to verify the disinformation, which, along with the claim, are sent to the LLM for final verification.

For RQ2, the pipeline is shown in the third row of Figure 2. Instead of self-generated CoT process, the models are prompted to follow the fact-checking (FC) pipeline to verify the claims in the following process: evidence retrieval, verdict prediction, and justification production.

¹https://www.reuters.com/

²The knowledge cutoff dates of GPT-40 and Gemini-1.5-flash is October 2023 and November 2023, respectively.

TABLE I: Reuters verdicts labels. Reuters fact-check articles assign a verdict to the central claim and we categorize these verdicts into true and false.

Туре	Numbers	Publish Date Range	Reuters Fact-check Verdicts	Standard Labels
News	73	Feb 2024 - Aug 2024	True	True
Disinformation	40	Feb 2024 - Aug 2024	Misleading, Missing Context, Altered, Synthetic Media, Miscapthioned, Satire	False

TABLE II: Experimental results of empirical study.

Approach	Veri	fication Rate	Sı	iccess Rate
	GPT-40	Gemini-1.5-flash	GPT-40	Gemini-1.5-flash
Zero-shot	6.3%	4.5%	2.7%	1.8%
CoT	9.8%	7.4%	5.3%	2.6%
FC guidance	20.4%	91.2%	17.7%	71.7%
With evidence	100%	100%	90.3%	92.9%

For RQ3, the pipeline is shown in the last row of Figure 2. This is a semi-automatic evaluation strategy that we first manually collect human-written evidence from the fact-check articles, which are expected to be highly relevant to the claims, then add the supporting evidence into the context, and submit the whole context to the LLMs for verification.

For all methods, we analyze LLM's responses in three aspects: (1) whether the LLM can perform the verification task; (2) what is the success rate if the LLMs can execute the task; (3) if there are any errors in the responses, e.g., hallucinations, that can affect the verification accuracy and reliability. This evaluation has two benefits. First, we can directly observe LLMs' performance in verification tasks under the different approaches; Second, manual evaluation allows us to further explore what the challenge is if LLMs cannot verify the truthfulness of disinformation.

- *3) Evaluation Metric:* To ensure reliability and validity, we repeat each evaluation 5 times, resulting in a total number of 1130 trials (i.e., 2 models * 113 examples * 5 repetitions). We use the following metrics to benchmark the capability and detection accuracy of LLMs.
- **Verification rate:** This measures the confidence of LLMs in claim verification. If the LLMs respond with "Yes, I can do" or output the prediction verdict with justification, we consider that it possesses this ability. Otherwise, outputs like "No, I cannot do" denote that models cannot complete the task.
- Success rate: This assesses the ratio that LLMs correctly predict the verification result.

B. (RQ1) Unsupervised Verification

We first explore whether existing LLMs can be directly utilized to verify the truthfulness of claims solely based on the content without human involvement, and if not, what the key obstacles are.

The experimental results are shown in Table II and some representative examples are shown in Figure 3. From the success rate of claim verification with zero-shot or CoT, it is obvious that neither of them can complete the task effectively.

Gemini-1.5 only achieves a 1.8% success rate through zero-shot prompting and a 2.6% success rate through CoT. For GPT-40, although CoT performs better than zero-shot, the success rate is still very low (5.3%).

We further analyze the LLM responses to understand why they sometimes fail to complete the verification task. The LLMs utilize the CoT approach to generate a series of general steps aimed at verifying the claim from various perspectives. However, they often struggle to access specific information relevant to these steps and the target claim. We present two concrete examples in Figure 3. In case #1, GPT-40 aims to validate the claim using its own text and image through zero-shot prompting. However, it answers that it cannot verify the claim without additional information. In case #2, GPT-40 generates 4 steps to verify the claim through CoT, as shown in answer 1. However, it fails because it does not find additional information related to these 4 aspects, as shown in answer 2.

In summary, due to the lack of sufficient context and specific information, LLMs are unable to generate accurate judgments on the claim independently. Consequently, LLMs frequently fail to verify the truthfulness of disinformation, particularly for claims that have emerged after the cutoff date of their training data. This highlights a critical challenge in relying solely on LLMs for disinformation verification in dynamically evolving information environments.

Finding 1: *LLMs CANNOT accurately verify the truth-fulness of the claim directly.*

C. (RQ2) Verification with Fact-checking Guidance

We further assess the LLMs' capability of detecting disinformation with the fact-checking guidance provided by humans. The results are shown in Table II ("FC guidance" row). We analyze the results from the verdict and justification generated by the LLMs, respectively.

Verdict analysis. We observe that the performance differences between GPT-40 and Gemini-1.5-flash in this scenario become pronounced. GPT-40 achieves a verification rate of 20.4% and success rate of 17.7%, while Gemini-1.5-flash achieves a verification rate of 91.2% and a success rate of 71.7%. However, the high success rate of Gemini-1.5-Flash is misleading, as it does not accurately and comprehensively reflect the model's actual performance and limitations. We will explain the reason later in the justification analysis. After analyzing the answers of the LLMs, the low verification rate of GPT-40 is led by that GPT-40 considers it cannot retrieve information from the web (as exemplified by *case #5* in Figure 3), while Gemini-1.5-flash only outputs the same answer encountering a small

part of cases (9.8%). This directly leads to the great gap of verification success rate between GPT-40 and Gemini-1.5-flash. These two LLMs predict the claims with true labels more accurately than the claims with false labels.

Justification analysis. To ensure the reliability of the verdict, we analyze the justifications generated by the LLMs. When guided through the FC processes, the LLMs' justification will contain the evidence that is their reference for judging and the source link of the evidence. We meticulously check the evidence provided by LLMs. Among the outputs containing both verdicts and justifications, we calculated the proportion of justifications that included links. For GPT-40, the proportion is 45.5%, while for Gemini-1.5-flash, it is 50.0%. An example of the evidence is shown in Figure 3, case #6, answer 1 (we just present links here and highlight them in red). However, almost all of the links are not accessible, even if they seem correct. The rest of the links located on the websites are irrelevant to the evidence. Since the LLMs are unable to search for information from the Internet, and the fact-check samples are published after the cut-off date of the LLMs, this evidence is potentially generated by hallucination. Therefore, despite the success rate of Gemini-1.5-flash seeming relatively high, it compromises our trust in its verdict. Since the evidence is retrieved from unverified web pages, the LLMs may make the correct prediction by coincidence instead of finding credible and relevant evidence and then reasoning from it. In real-world fact-checking, we do not know the ground-truth label of a claim in advance, thereby we cannot judge the verdict. The inaccessible reference links in the LLMs' response definitely make the verdict unconvincing to humans and cause suspicion to a certain extent.

Based on the above analysis, we conclude that LLMs' cannot retrieve evidence from the Internet, and the evidence can be potentially generated by their hallucinations with inaccessible source links. This undermines users' trust in applying LLMs to disinformation detection.

Finding 2: LLMs have shortcomings in searching for claim-relevant public information and their responses may include hallucinated links that weaken result trust-worthiness.

D. (RQ3) Verification with Human-written Evidence

We evaluate whether providing additional evidence written by humans could enhance LLMs' capability of claim verification. The evaluation results are presented in Table II ("With evidence" row). We observe that LLMs demonstrate relatively high performance in the verification when providing external evidence. GPT-40 and Gemini-1.5-flash achieve success rates of 90.3% and 92.9%, respectively. Compared to the RQ1 results, providing LLMs with human-written evidence significantly boosts their performance. The success rates of GPT-40 and Gemini-1.5-flash improve by 85.0%, and 90.3%, respectively.

Failure case analysis. Then, we manually analyze the failure cases by examining their verdicts and justifications. The failure

reasons can be summarized into two types. The first is that the verdicts and justifications are logically inconsistent. This indicates a mismatch between the verdict and the justification: while the justification aligns with the ground truth (e.g., providing evidence that refutes a false claim), the predicted verdict contradicts it (e.g., predicting true). An illustrative example is case #3 in Figure 3: the ground-truth label of the claim is false, and GPT-40 generates a justification based on the evidence that refutes the claim. However, it mistakenly predicts the claim to be true. The second is hallucinations in justifications. There is 1 justification from GPT-40 and 2 justifications from Gemini-1.5-flash that contain hallucinations. This indicates the LLMs wrongly summarize the provided evidence and the generated justifications distorte the original meaning of the evidence. An example is case #4 in Figure 3, where hallucinations generated by Gemini-1.5-flash are highlighted in red. The LLM fabricates non-existent facts based on the original evidence, leading to a justification that conveys a different meaning from the original evidence.

From the above analysis, we conclude that LLMs demonstrate exceptional fact-check capabilities when supported by human-written evidence. Although there are a few minor flaws, they do not overshadow the overall strengths. LLMs can generate reasonable justifications by summarizing the evidence to support their verdict and thus correctly verify the claim. The reliable evidence improves users' understanding and trust in the prediction results.

Finding 3: Human-written evidence enhances LLMs' ability to verify multimodal claims and generate coherent justifications.

V. HOLMES

Based on the above findings, the enabling factor for accurate LLM-based misinformation detection is to enhance their capabilities of retrieving high-quality and relevant evidence from the public Internet. Inspired by this, we introduce Holmes, an end-to-end framework to automatically check and verify multimodal claims with contemporary LLMs. The core of Holmes is a novel, enhanced evidence retrieval methodology. It harnesses the summarization capability of LLMs to extract public information and designs a new algorithm and metric to evaluate the quality of extracted evidence. This ensures the collected evidence is highly relevant and accurate and does not contain any hallucinations, significantly improving the accuracy of the verification.

Figure 4 shows the overview of Holmes. It first decomposes the original multimodal claim into separate single-modal sub-claims (e.g., text and image). Then it performs the following two main phases: (1) *Evidence Retrieval*: Holmes invokes a third-party search engine to search for information related to each sub-claim, and leverages the component of distilling LLMs to precisely extract the key points from the bulk of original information as evidence. (2) *Verification and Justification*: Holmes uploads the original claim and summarized evidence together to the LLM and guides it to

Claim (Text)	Claim (Image)	LLM	Strategy	LLM's Answers	Analysis
#1. Scenes from Sheikh Hasina's Bedroom.	Jan de la constitución de la con	GPT-4o	Zero-shot prompting	I cannot verify the claim Verifying such a specific claim would require additional context or authentic sources confirming the location and scenario depicted in the image.	Lack in additional related information.
#2. Kevin Costner Refused to be on the Same Stage with Whoopi Goldberg at the Oscars "Not if my life depend on it ".		GPT-4o	СоТ	Answer 1:Sure, I can help you verify the claim step by step. 1. Source Verification: 2. Cross-Referencing: 3. Official Confirmation: 4. Recent Updates: Answer 2: The is not verifiable with the information provided.	CoT method fails for lacking in additional related information.
#3. It's official. Fox is completely out of bullshit.		GPT-4o	With evidence	True. 1. A spokesperson for Fox News confirmed that the screenshot was digitally altered	Logical inconsistence between verdict and justification.
#4. It's the end of an era, Cartoon Network has officially shut down. Here are all the shows that made our childhood! My God, I really miss Bob the builders, The Koala Brothers, Franklin, Andy Pandy from my times!	SALUSCO A. J.	Gemini- 1.5-flash	With evidence	False. 1 2. The campaign using #RIPCartoonNetwork was aimed at bringing attention to the challenges facing animation workers, particularly job losses due to outsourcing and corporate mergers, rather than indicating the network's closure	Hallucinations exist in the justification.
#5. Secret Service Director kimberly Cheatle has been ARRESTED.		GPT-4o	FC guidance	No. I could not retrieve any relevant evidence related to the claim.	LLMs cannot retrieve evidence.
#6. Breaking: ABC Signs A \$400 Million Deal With Tucker Carlson For A Late- Night Show, "Going to Replace Jimmy Kimmel Show".		Gemini- 1.5-flash	FC guidance	Answer 1: 1. https://www.snopes.com/fact-check/tucker-carlson-abc-deal/ 2. https://www.thewrap.com/tucker-carlson-abc-deal-fake-news/ Answer 2: False. 1. The article claims that Tucker Carlson is going to replace Jimmy Kimmel on his late night show	The source links of the evidence searched by LLMs are not accessible.

Fig. 3: Representative examples in our empirical study. *Analysis* denotes the manual analysis of the failure reason. The incorrect contents of the answers generated by the LLMs are highlighted in red.

generate the verdict and justification in a certain format after checking the truthfulness of the claim. Below we give the detailed design of each phase.

A. Evidence Retrieval

The goal of this phase is to search for the public information relevant to each sub-claim and extract it as the evidence. Note that we collect the evidence for each modality separately, as this can bring us more comprehensive sources, and current search engines [46], [47], [48] are less powerful to simultaneously search for multi-modalities. The retrieval consists of the following three steps: (1) it locates the credible sources where the sub-claims appear; (2) it extracts the main content from the located web pages; (3) it evaluates the extracted content and selects the best one as evidence. Below we give details of each step.

1) Locating Sources: As the first step, Holmes locates the web pages whose contents are highly relevant to each subclaim. To achieve this goal, it integrates a text search engine and an image reverse search engine for the two sub-claims, respectively. The text search engine responds to the sources of the document or web pages relevant to the text sub-claim, while the image reverse search engine identifies the sources of the image sub-claim or the sources of similar images.

In practice, Holmes uses Google Custom Search Engine [46] for the text domain and Google Reverse Search Engine [49] for the image domain. It submits the text and image sub-claim to the corresponding search engine APIs. Google's text-based search engine returns web links whose content is semantically or contextually similar to the input

query. In contrast, Google's reverse image search engine identifies and traces the origins of images by returning web links that contain matching or visually similar images. Holmes parses the search results, and extracts the webpage links in sequence. In this stage, Holmes builds a *text-based evidence links list* and an *image-based evidence links list*.

2) Content Extraction: In this stage, Holmes leverages a Selenium-based crawler to extract textual content from web pages listed in the text-based evidence links and image-based evidence links. Selenium [50] automates webpage interactions, effectively simulating human browsing behaviors to capture comprehensive webpage content. However, raw extracted content often includes irrelevant elements such as privacy policies, advertisements, and other extraneous information unsuitable as evidence. To address this issue, we utilize the Python package newspaper3K [51], which efficiently identifies and extracts the primary textual content from webpages, excluding advertisements, headers, footers, and miscellaneous text. It is noteworthy that, due to anti-crawling measures implemented by certain websites, Holmes occasionally encounters access restrictions. In these scenarios, we employ Google's brief summary snippets of the inaccessible websites as a proxy for their original textual content.

Then Holmes initializes an LLM to extract valuable content from the original information. LLMs have been used for summarization of articles or long paragraphs [52]. Inspired by the fake news detection policy [53] that introduces how to fact-check suspicious information and retrieve relevant information, we meticulously design a prompt template to offer comprehensive guidance for the LLM to gather com-

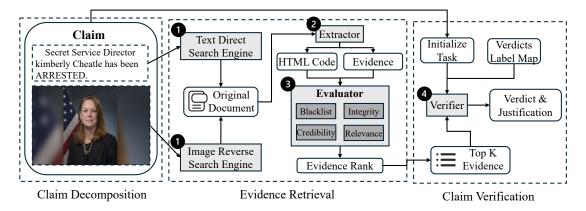


Fig. 4: Overview of Holmes. (1) It decomposes the claim into text and image sub-claims and searches for the information related to each sub-claim. (2) It uses an LLM to summarize the original document into evidence and then evaluate the quality of the evidence. (3) It assigns the verification task and submits the high-quality evidence to the LLM, which will output the verdict for the claim and justification to support its verdict.

plete, detailed, and structured information, ensuring that the original text is thoroughly investigated and well-understood. This prompt template instructs the LLM to investigate eight fundamental elements consisting of an article: **People**, **Event**, **Location**, **Time**, **Reason**, **Background**, **Impact** and **Followup**. The prompt is shown in Appendix C.

- 3) Evaluating Evidence: It is critical to assess whether the extracted evidence is qualified for the claim verification. Holmes evaluates the evidence from three angles: credibility, relevance, and integrity. It filters out the evidence from sources with low credibility, and grades the rest by auditing its relevance to the claim and integrity of the event extraction. Given a claim, let C denote its semantic representation and the set E denotes the evidence extracted. It consists of the pairs $(e_i, link_i)$ indicating the text evidence and its source link. Details of the metrics are given below:
- Credibility. This metric is essential for verifying the reliability of the evidence. First, Holmes uses a blacklist [54], [55] to exclude evidence from low-credibility websites and bias domains. Next, It applies a machine learning model [56] to assess the credibility of remaining webpages. Evidence from pages that meets a predefined credibility threshold is retained, enhancing the quality standard of the evidence. This is formulated as follows:

$$Cred(link_i) = \begin{cases} 1, & \text{If the link is credible.} \\ 0, & \text{If the link is not credible.} \end{cases}$$
 (1)

Relevance. This metric quantifies the similarity between
the evidence and the claim. Holmes deploys BLIP-2[57]
to generate a semantic representation of the claim and then
calculates the cosine similarity between this representation
and the evidence text to derive a relevance score. A higher
score indicates a stronger alignment between the evidence
and claim. This is formulated as follows:

$$Rel(C, e_i) = cos \ sim(C, e_i)$$
 (2)

• Integrity. This metric assesses the coverage of event arguments in the evidence. Holmes deploys ChatIE[58] to extract events from the evidence and gets a structured table that includes the argument role and argument content. If each argument role has the corresponding content, it is considered as "filled"; if there is no content (represented as None), it is considered as "unfilled." The integrity score is defined as the proportion of filled arguments. This is formulated as follows:

$$Int(e_i) = \frac{\# \ of \ Filled \ Arguments \ in \ e_i}{Total \ \# \ of \ Arguments \ in \ e_i}$$
(3)

Algorithm 1 shows the detailed process of evaluating the quality of the evidence. Given a claim, its semantic Representation C and its related evidence set E, Holmes first checks if $link_i$ is creditable and not in the blacklist. Then it computes the *Relevance* and *Integrity* scores of each e_i , as well as the evidence score $EQ(e_i)$ as below:

$$EQ(e_i) = \alpha \cdot Rel(e_i, C) + (1 - \alpha) \cdot Int(e_i), \tag{4}$$

where the hyperparameter α is to balance the weight of these two metrics, and we set it as 0.5. Other hyperparameter settings and the experiment results are provided in Appendix A. Finally, Holmes sorts the evidence pairs in set E by their $EQ(e_i)$ values.

Our evidence retrieval methodology has two advantages. First, it is able to collect abundant related and reliable information from open source; Second, it distills the long paragraphs into clear and concise evidence which can relieve the pressure for the LLM to analyze and interpret.

B. Claim Verification

In the second phase, Holmes guide the LLM to verify the truthfulness of the claim and generate their justification using the retrieved evidence. To enhance the reasoning performance of the LLM, we divide this phase into three steps which are executed sequentially. This can reduce the token length of

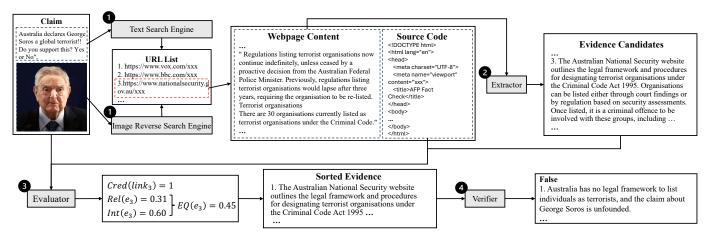


Fig. 5: An illustrative example of how Holmes automatically verifies the real-world multimodal misinformation.

```
Algorithm 1: Evidence Evaluation Algorithm

Input: Claim Semantic Representation C, Evidence Set E = \{(link_1, e_1), (link_2, e_2), \dots, (link_n, e_n)\}
Output: Set E with Related Quality Score

1 Evaluate Stage:
2 for (e_i, link_i) in E do

3 | if link_i not in blacklist and Cred(link_i) = 1 then

4 | Calculate Rel(e_i, C);

5 | Calculate Int(e_i);

6 | EQ(e_i) = \alpha \cdot Rel(e_i, C) + (1 - \alpha) \cdot Int(e_i);

7 Sort E by EQ(e_i) in descending order;

8 return E
```

each query and take advantage of the ability to memorize the context of the LLM. The three steps are shown below:

- Initialization. Holmes assigns the verification task to the LLM and submit the claim to it. Note that the LLM does not need to execute any reasoning operation in this step, except for memorizing the task and the claim. The prompt template is shown in Appendix C.
- **Uploading evidence.** Holmes uploads the retrieved evidence to the LLM. It adds the evidence into its context and waits for the next instruction. The prompt template is shown in Appendix C.
- **Verifying the claim.** Holmes instructs the LLM to use the evidence to verify the truthfulness of the claim and output the prediction and explanation reasons in the required format. The prompt template is shown in the gray box below. The {true_verdict_set} and {false_verdict_set} are shown at the top and the bottom of Table III, column 2. The {output format} is shown in Appendix C.

For verdict, Holmes simplifies the task into binary classification. To be specific, for any subcategory classification (shown in Table III) made by the LLM, the result will ultimately be mapped to either true or false. For example,

TABLE III: The verdict label mapping used in this paper, which is collected from fact check agents.

Standard Labels	Fact Check Agent Labels				
True	Accurate, Mostly-Accurate, Correct, Partially-Correct, Mostly correct, Partially True, Mostly True, True				
False	Misleading, Missing Context, Altered, Synthetic Media, Miscapthioned, Satire, Fake News, Inaccurate, Incorrect, Likely False, Misrepresented, Missing Context, Mostly False				

if the claim is a satire, Holmes labels it as false. This rule offers several key advantages. First, it reduces the complexity of the decision-making process for the LLM, allowing it to focus on the fundamental task of claim verification without being distracted by fine-grained subcategory classification. Second, this enhances LLM's generalization and ensures the consistency of different criteria for classifications from various fact-checking agents. Third, binary classification makes the results more straightforward and user-friendly, as end users often care primarily about the truthfulness of the information rather than its specific category.

To enable Holmes to detect more types of disinformation and overcome the issue that disinformation labels are adjusted by fact-check articles over time, we supplement the labels in Reuters (shown in Table I) by collecting the labels mentioned in other references and fact check agents [59], [60]. In the building process, we avoid some labels overlapping with others and collect labels with limited examples. The complete version is shown in Table III. It contains 8 true verdicts and 13 false verdicts. Finally, Holmes guides the verification LLM to generate the justification of its verdict in the required form. Through referring to the justification written by Reuters, we design a user-friendly and lucid justification template as shown in Appendix C.

The justification provides explainability of LLM's verification process with the following benefits. First, it strengthens the credibility of the verdict, making it more convincing and reliable. Second, it enhances transparency and trust that mitigates potential disputes by offering clear reasoning.

C. An Illustrative Example

We use a real-world disinformation example to illuminate how Holmes automated checks the claim's truthfulness, as shown in Figure 5. The claim states that "Australia declares George Soros a global terrorist!! Do you support this? Yes or No" with a portrait of George Soros. The fact-checking process has four stages. • Holmes searches the sources of the text and image of the claim on the Internet, respectively, and obtains 20 links in total. 20 Then Holmes crawls the webpage content and the source code of these links and summarizes the precise abstract of the webpage content. These 20 summaries are evidence candidates that await quality evaluation. We take candidate 3 whose source is link 3 (https://www.nationalsecurity.gov.au/xxx) as an example. Its main content is that "The Australian National Security website outlines the legal framework and procedures for designating terrorist organisations under the Criminal Code Act 1995". 3 The value of credibility of link 3 is 1, indicating it is credible. The evidence quality score of evidence candidate 3 is 0.45. Depending on the score, evidence candidate 3 becomes the first evidence candidate with the highest score. 4 Holmes uses the top 5 pieces of evidence in the sorted evidence set to verify the claim and outputs the verdict and the justification. Holmes verifies the claim as false and the justification generated from the example evidence is "Australia has no legal framework to list individuals as terrorists, and the claim about George Soros is unfounded.".

VI. EVALUATION

We evaluate the performance of our method on the multimodality disinformation datasets. In particular, we focus on the following three questions.

- RQ1 (Verification with Proofs) How effective is Holmes in verifying disinformation on public datasets that include supporting proofs?
- **RQ2** (Verification with Search) Can Holmes verify disinformation by searching for evidence autonomously?
- **RQ3** (**Ablation Study**) How effective are each of the components within the proposed methodology?

A. Experimental Setup

Datasets. We evaluate Holmes using three datasets, with detailed statistics presented in Table IV. Two of these are publicly available multimodal disinformation datasets from prior research: Mocheg [21] and MR2 [20], which are used to assess **RQ1**. Samples in Mocheg consist of textual claims, multimodal supporting evidence, and labels, whereas those in MR2 include multimodal claims, multimodal supporting evidence, and labels. Notably, we filter out the samples labeled as NEI (Not Enough Information) and unverified among Mocheg and MR2 datasets, respectively, due to RQ1 being a binary classification task. Finally, we obtained 8,368 training samples and 1,642 test samples for the Mocheg dataset and 2,988 training samples and 802 test samples for the MR2 dataset.

To evaluate Holmes for **RQ2** and **RQ3**, we construct a new dataset named the Multi-Source Multimodal Disinformation

TABLE IV: Samples statistics of the benchmark datasets. Positive samples and negative samples denote true and false information, respectively. X indicates the benchmark does not have this set.

	# Positiv	e Samples	# Negati	ve Samples	Lables
	Train	Test	Train	Test	
Mocheg	3,826	817	4,542	825	supported, refuted
MR2	1,854	411	1,134	391	non-rumor, rumor
MMDV	X	609	X	605	true, false

TABLE V: Comparison of baselines models. *Multimodal* denotes the ability to detect multimodal disinformation; *Evidence Retrieval* denotes the ability to search information from the Internet; *Explainability* denotes the ability to generate justification for detection results.

	Multimodal	Evidence Retrieval	Explainability
Pre-CoFactv2	✓	Х	Х
End-to-End	✓	X	✓
MR2	✓	✓	×
SpotFakePlus	✓	×	×
Holmes	✓	✓	✓

Verification Dataset (Abbreviated as *MMDV*), as the two open-source datasets used in RQ1 are not suitable for this purpose. In this setting, Holmes is required to verify disinformation without access to predefined supporting evidence. While we can evaluate Holmes on these two datasets without providing evidence, the results may be unreliable. This is because the two datasets were created before the knowledge cutoff dates of GPT-40 and Gemini 1.5 Flash, and the relevant information may already exist in the training data of these models. Consequently, Holmes could potentially verify disinformation based on memorized knowledge rather than retrieved external evidence. To eliminate this uncertain factor, we construct a new dataset specifically tailored for this evaluation scenario.

The MMDV dataset is constructed following the two fundamental requirements: (1) The dataset consists of claims and ground truth labels, excluding any supporting evidence. (2) The claims are published later than the training cutoff date of GPT-40 and Gemini-1.5-flash, ensuring a fair evaluation. The samples in the MMDV dataset are collected from three fact check agents: Snopes³, PolitiFact⁴, and Reuters. The construction process of the MMDV dataset is detailed as follows. For samples from Snopes and PolitiFact, we collect text claims, associated images, and rates from the fact-check articles and annotate the samples with binary labels (true or false) based on their fine-grained rates provided in the article following the rule in the reference [21]. For samples from Reuters, we adopt a different strategy: (1) We collect positive samples from the Reuters News column. To ensure variety and randomness, we select an equal number of articles from three categories—World, Business, and Markets—and

³https://www.snopes.com/

⁴https://www.politifact.com/

use the title of each article as the claim. (2) We collect negative samples from the Reuters Fact Check column, where articles explicitly identify the sources of disinformation. (3) Finally, positive samples are labeled true, and negative samples false. We have open-sourced our dataset at our anonymous: https://zenodo.org/records/15275006, for a fair evaluation.

Baselines. We incorporate two state-of-the-art commercial multi-modality LLMs into Holmes: OpenAI GPT-40 and Google Gemini-1.5-flash. We also implement Holmes with two open-source LLMs: Llama 3.2-vision-11B [61] and Qwenvision-7B [62] as the alternative choice. For benchmark methods, we select four SoTA solutions: End2End [21], Retrieval-Based [20], Pre-CoFactv2 [23], and SpotFakePlus [22]. The characteristic details of Holmes and baselines are shown in Table V. All these methods are designed for multimodal verification tasks that can handle multi-modality claims and evidence as inputs, but differences still exist. Among the baseline methods, only the MR2 method can automatically retrieve external evidence from open sources without manual collection. Notably, although the End-to-End method is capable of retrieving evidence from a closed-source knowledge base, the knowledge base must be manually constructed in advance. As such, all evidence is pre-curated and embedded within the knowledge base beforehand. The method merely selects relevant evidence from this fixed pool, rather than autonomously retrieving and synthesizing evidence from open sources such as the Internet. Therefore, it does not meet our criteria for evidence retrieval. Moreover, among the baselines, only the End-to-End method is capable of generating justifications to explain its predictions, while other methods provide only classification results. In contrast to existing baselines, which lack one or more key capabilities, Holmes uniquely integrates open-source evidence retrieval, disinformation verification, and justification generation, fully aligning with the goals of automated fact-checking.

Settings. To evaluate the performance of Holmes and the baseline models in verification tasks, we adopt four standard metrics commonly used in binary classification: Accuracy, Precision, Recall, and F1-score. In order to replicate the baseline models as faithfully as possible, we follow their specified versions of Python and dependencies, and deploy them on a server running Ubuntu 18.04.6 LTS, equipped with an NVIDIA GeForce RTX 3090 GPU (24 GB VRAM). We implement four variants of Holmes: GPT-40, Gemini-1.5-Flash-001, Llama 3.2-vision-11B and Qwen-vision-7B respectively as both the extractor and verifier LLMs in Holmes as illustrated in Figure 4. Notably, for each variant of Holmes, the same model is used for both components. We access the commercial LLMs via API endpoints whereas the open-source models are run on NVIDIA GeForce RTX A6000 GPU with 48GB of VRAM. The temperature of the LLMs is set to 0 to ensure stable outputs.

B. (RQ1) Verification with Proofs

We first investigate the performance of Holmes when it performs verification with provided proofs, and compare it

with the four baseline models detailed in the Experimental Setup. The evaluation strategies include two aspects: (1) Performance Evaluation: Each baseline model is trained and evaluated separately on the Mocheg and MR2 datasets using their respective train-test splits. For Holmes, we evaluate it on the test set of Mocheg and MR2 without a training process. (2) Transferability Evaluation: For each baseline model, we train it on the training set of Mocheg or MR2 and evaluate its performance on the test set of the other. This experimental setup is designed to simulate real-world deployment scenarios, where the sources of disinformation are diverse and not confined to a single domain.

The performance evaluation results are presented in Table VI. In general, Holmes performs better than the baseline models(Pre-CoFactv2, End-to-End, MR2, and SpotFakePlus), excluding Holmes (Llama 3.2-vision) on the two benchmarks. Among the baselines, the original benchmark and baselines, evaluated on their native datasets, achieve the best performance(e.g., Mocheg & End-to-End and MR2 & MR2). This may be attributed to the alignment between the model and its native dataset. but their accuracy is still lower than Holmes. When deployed with open-source LLMs(Llama 3.2-vision and Qwen-vision), Holmes performs less than when deployed with commercial LLMs(Gemini-1.5-flash and GPT-4o). Llama 3.2-vision performs badly on the MR2 dataset. We conclude that it has poor ability to handle multi-images by analyzing the response. Notably, Holmes deployed with GPT-40 outperforms any other approaches across the two benchmarks, achieving the highest accuracy (73.8% on Mocheg and 88.3% on MR2), precision (75.9% on Mocheg and 88.8% on MR2), recall (73.9% on Mocheg and 88.2% on MR2), and F1 score (73.2% on Mocheg and 88.3% on MR2). This indicates the superior capability of Holmes in this fact-checking task.

The transferability evaluation results in Table VII show that Holmes achieves the best performance deployed with various LLMs which all evaluation metrics are higher than the four baseline models Pre-CoFactv2, End-to-End, MR2, and SpotFakePlus. Notably, GPT-40 continues to outperform all other models across both datasets, achieving the highest accuracy. For baseline models that require training, there is a noticeable drop in performance compared to when they are trained and tested on the same dataset. For example, Pre-CoFactv2 sees a significant decrease in its F1 score, dropping to 23.3% on Mocheg and 32.2% on MR2. Similarly, End-to-End and MR2 show large declines in all metrics, with MR2 performing particularly poorly on Mocheg (19.2% F1 score). SpotFakePlus also shows a marked decrease in its metrics, especially with precision and F1 scores in both datasets. In contrast, Holmes, the zero-shot approach that does not require training maintains performance, showing leading advantages in this test. This suggests that Holmes has good transferability and robustness capable of verifying various disinformation.

C. (RQ2) Verification with Evidence

We then explore the performance of Holmes in verifying disinformation without predefined proofs, requiring the

TABLE VI: The *performance* comparison on the benchmark datasets. The baseline models in the first column are trained and evaluated on the Mocheg or MR2 benchmark datasets.

	Mocheg				MR2			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
Pre-CoFactv2	46.7%	51.1%	46.3%	41.1%	60.2%	64.0%	57.5%	57.2%
End-to-End	54.5%	55.8%	54.2%	51.7%	54.4%	55.9%	54.1%	51.8%
MR2	37.5%	44.6%	37.0%	25.6%	62.8%	66.6%	60.3%	59.2%
SpotFakePlus	53.0%	54.8%	54.7%	52.9%	54.0%	55.6%	54.7%	52.2%
Holmes (Llama 3.2-vision)	61.1%	62.4%	61.1%	60.0%	34.9%	17.4%	50.0%	25.9%
Holmes (Qwen-vision)	47.3%	40.3%	47.4%	34.6%	66.2%	34.6%	46.9%	39.8%
Holmes (Gemini-1.5-flash)	64.9%	65.1%	64.9%	65.0%	73.8%	63.1%	74.7%	68.4%
Holmes (GPT-4o)	73.8%	75.9%	73.9%	73.2%	88.3%	88.8%	88.2%	88.3%

TABLE VII: The *transferability* comparison on the benchmark datasets. Baseline models are trained on one dataset and evaluated on another. For example, in the first row, Mocheg (MR2) indicates that the models are trained on the MR2 training set and evaluated on the Mocheg test set. The ↓ indicates that this metric has decreased compared to Table VI.

	Mocheg(MR2)				MR2(Mocheg)			
	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
Pre-CoFactv2	34.2% ↓	32.8% ↓	33.8% ↓	23.3% ↓	34.6% ↓	36.0% ↓	36.8% ↓	32.2% ↓
End-to-End	36.2% ↓	38.6% ↓	38.7% ↓	29.2% ↓	35.5% ↓	37.6% ↓	37.9% ↓	28.8% ↓
MR2	33.5% ↓	29.3% ↓	33.1% ↓	19.2% ↓	34.2% ↓	41.9% ↓	36.2% ↓	29.0% ↓
SpotFakePlus	34.3% ↓	29.6% ↓	34.8% ↓	20.9% ↓	36.5% ↓	23.8% ↓	36.4% ↓	28.1% ↓
Holmes (Llama 3.2-vision)	61.1%	62.4%	61.1%	60.0%	34.9%	17.4%	50.0%	25.9%
Holmes (Qwen-vision)	47.3%	40.3%	47.4%	34.6%	66.2%	34.6%	46.9%	39.8%
Holmes (Gemini-1.5-flash)	64.9%	65.1%	64.9%	65.0%	73.8%	63.1%	74.7%	68.4%
Holmes (GPT-4o)	73.8%	75.9%	73.9%	73.2%	88.3%	88.8%	88.2%	88.3%

TABLE VIII: The experimental results for RQ2. Note that the first three methods are unable to retrieve evidence from the Internet.

	Accuracy	Precision	Recall	F1
Pre-CoFactv2	41.9%	0.0%	0.0%	0.0%
End-to-End	41.0%	20.5%	50.0%	29.1%
SpotFakePlus	27.0%	17.9%	30.6%	21.8%
MR2	13.0%	16.4%	10.3%	12.7%
Holmes (Llama 3.2-vision)	73.4%	70.8%	56.0%	48.1%
Holmes (Qwen-vision)	71.1%	39.6%	43.4%	41.6%
Holmes (Gemini-1.5-flash)	87.0%	87.8%	86.7%	86.8%
Holmes (GPT-4o)	90.2%	89.9%	90.0%	89.8%

method to actively search for supporting evidence. This task necessitates that the detection models independently complete the verification task when encountering an unverified claim without any human-provided additional evidence. Specifically, the evaluation strategy is as follows: If the models can retrieve evidence themselves, they combine the claim and the retrieved evidence to verify the truthfulness of the claim. Otherwise, the models verify the claim based solely on its content. Based on this strategy, Pre-CoFactv2, End-to-End, and SpotFake cannot retrieve evidence from the Internet, while MR2 and Holmes, which can, are called retrieval-based approaches.

The results are presented in Table VIII. Overall, evidence-based approaches outperform content-only methods, highlighting the critical role of evidence retrieval in verification accuracy. Methods relying solely on claim content, such as Pre-CoFactv2, End-to-End, and SpotFakePlus, achieve poorer results. MR2 records the lowest accuracy (13.0%), reflecting its limited transferability. Holmes significantly outperforms baselines, achieving the highest accuracy (85.1%) with GPT-40 and strong performance (84.1%) with Gemini-1.5-flash.

Additionally, Holmes using open-source models Llmma 3.2-vision and Qwen-vision surpasses other baselines but remains slightly behind commercial LLM models. These outcomes demonstrate the substantial advantage of Holmes's evidence retrieval capability in real-time disinformation detection. The cost of Holmes for verification is shown in Section B.

D. (RQ3) Ablation Study

To investigate the effectiveness of our evidence-retrieval approach, we conduct experiments to compare the evidence retrieval method in Holmes with that used in the previous study [24], [20]. We use MR2 to indicate this evidence retrieval method. The detail of MR2 is as follows: It initialize a crawler that first uses Google Reverse Image Search to collect textual evidence by crawling descriptions of similar images. Then the crawler identifies image tags, extracts descriptions from <figcaption> and image-related attributes (e.g., <alt>, <caption>), and compiles non-redundant text snippets from each web page for analysis. Additionally, visual evidence is retrieved using the Google Programmable Search Engine with the text of the post as the query, retaining the top 5 images after filtering disinformation sources. We set the MMDV dataset as the benchmark of this experiment and evaluate Holmes deployed with four LLMs: GPT-4o, Gemini-1.5-flash, Llama 3.2-vision, and Owen-vision.

The comparison results are shown in Table IX. Keeping other settings identical but with different evidence retrieval methods, Holmes performs better in detecting disinformation when using our evidence retrieval method than when using the MR2 method. The main difference of the two approaches is that we extract the main content of the original web pages, while the MR2 method collects partial text in HTML tags as

TABLE IX: Experimental results with different evidence retrieval approaches. The first row indicates the evidence retrieval approaches. The best metrics are bold for every LLM with different retrieval approaches.

	MR2				Holmes			
	Accuracy	Accuracy Precision Recall F1			Accuracy	Precision	Recall	F1
Holmes (Llama 3.2-vision)	56.3%	56.0%	54.9%	53.3%	73.4%	70.8%	56.0%	48.1%
Holmes (Qwen-vision)	47.2%	36.4%	45.0%	35.2%	71.1%	39.6%	43.4%	41.6%
Holmes (Gemini-1.5-flash)	65.3%	68.9%	66.1%	64.0%	87.0%	87.8%	86.7%	86.8%
Holmes (GPT-4o)	59.4%	65.1%	62.1%	54.8%	90.2%	89.9%	90.0%	89.8%

evidence, indicating that our method can obtain more abundant and comprehensive information to help LLMs more accurately verify the disinformation.

VII. DISCUSSION

A. Fact Check with LLMs with search capabilities

To enhance the reliability, accuracy, and currency of responses, OpenAI and Google have integrated online search functionality into their LLMs [63], [64]. Although not explicitly designed for fact-checking, this capability allows models to retrieve relevant online information, potentially supporting evidence-based verification tasks. Motivated by this potential, we empirically evaluated these search-enhanced models' performance in fact-checking tasks.

Setup. We selected three state-of-the-art LLMs: GPT-4o-search-preview [63], GPT-4o-mini-search-preview [63], and Gemini-1.5-flash-search-grounding [64]. The evaluation used the MMDV dataset with standard classification metrics: accuracy, precision, recall, and F1-score. Two limitations should be noted: First, since the MMDV dataset derives from publicly available fact-checking outcomes, embedded search engines might inadvertently retrieve these existing results. Current APIs lack domain-specific customization (whitelist/blacklist), complicating prevention of such access. Second, the selected LLMs do not support image modality [63], [64]. Hence, we utilized only the textual portion of multimodal claims. Models were required to provide a binary verdict (true/false) with coherent justifications.

Results. The results are shown in Table X. GPT-4o-search-preview achieves the highest accuracy of 88.3% among these 3 models. GPT-4o-mini-search-preview performs slightly worse with an accuracy of 85.3%. Gemini-1.5-flash-search-grounding achieves the lowest accuracy of 79.7%. Compared to Holmes, GPT-4o search-preview performs slightly worse on the verification task, with an accuracy that is 1.8% lower than Holmes (GPT-4o). Similarly, GPT-4o-mini underperforms by 4.7% compared to Holmes. The performance gap between Holmes (Gemini-1.5-flash) and Gemini-1.5-flash-search-grounding is bigger, with Holmes (Gemini-1.5-flash) achieving an accuracy that is 7.3% higher than that of Model Gemini-1.5-flash-search-grounding.

In our analysis of successful cases verified by SOTA LLMs equipped with search tools, we find that they can retrieve high-quality textual evidence to support claim verification, thereby achieving high accuracy. While these models are not specifically built for fact-checking tasks, it is possible that these enterprises maintain large-scale internal knowledge bases

TABLE X: The experimental results for LLMs with search function on MMDV dataset.

	Accuracy	Precision	Recall	F1
Gemini-1.5-flash-search-grounding	79.7%	80.9%	79.8%	79.5%
GPT-4o-search-preview	88.3%	88.4%	88.3%	88.2%
GPT-4o-mini-search-preview	85.3%	86.5%	85.5%	85.3%
Holmes (Gemini-1.5-flash)	87.0%	87.8%	86.7%	86.8%
Holmes (GPT-4o)	90.2%	89.9%	90.0%	89.8%

that enhance their performance. However, these methods still exhibit notable limitations in dealing with multimodal claims since their search functionality is restricted to textual content. Specifically, our analysis of failure cases reveals two common types of claims where performance drops significantly: (1) claims presented only by images; (2) claims with textual and visual content while the key information is conveyed mainly through visual content. Because these models are unable to retrieve or process visual evidence, they often fail to verify such claims. In contrast, Holmes is specifically designed to address the challenges of multimodal disinformation. It incorporates image reverse search tools to retrieve evidence relevant to the visual content, enabling it to capture key information that text-only systems overlook. As a result, Holmes demonstrates greater robustness and reliability in verifying complex multimodal claims.

B. Limitations and Future Work

Despite the effectiveness of Holmes in detecting disinformation across textual and image-based claims, it faces limitations when handling other modalities, such as audio or video. Detecting disinformation in these formats is especially challenging due to the complexity of temporal/visual-temporal signals, the need for synchronized multimodal reasoning, and the limited capabilities of current fact-checking frameworks in processing such content. Meanwhile, state-of-the-art tools like GPT-40 with web search or Gemini with Google Search primarily support textual input and lack robust support for audio-visual analysis. This reveals a critical blind spot in the current research: the absence of reliable systems for verifying multimedia content, where key evidence is probably embedded in non-textual formats. To address this gap, future work will integrate audio and video LLMs into Holmes to support broader and more robust multimodal fact-checking.

VIII. CONCLUSION

In this paper, we conduct a comprehensive study to investigate the ability of LLMs in verifying multimodal disinformation, disclosing the limitations of LLMs in achieving

this challenging goal. We further propose Holmes, a pioneering automated fact-check framework to detect multimodal disinformation. Holmes integrates a new evidence retrieval approach to acquire high-quality and relevant information from the public Internet, which can significantly improve the LLM's verification accuracy and rationality. Extensive experiments validate that Holmes significantly outperforms state-of-theart solutions over two multimodal benchmarks and a real-time dataset. We aim to design more complete solutions that can detect disinformation on other modalities as future work.

IX. ETHICAL STATEMENT.

This research adheres to the highest ethical standards in the development of automated fact-checking technologies. It is committed to verifying disinformation while respecting user privacy and ensuring data security. No personally identifiable information is collected or processed during the study, and all datasets used are either publicly available or properly anonymized in accordance with ethical guidelines. The objective of this work is to strengthen societal resilience against online disinformation by improving the accuracy and efficiency of multimodal fact-checking systems. Importantly, this research is strictly neutral and apolitical, conducted without any ideological or propagandistic intent. It upholds the principles of fairness, transparency, and integrity, aiming to provide an unbiased tool for identifying false information, independent of any narrative framing or institutional affiliation.

REFERENCES

- M. M. F. Caceres, J. P. Sosa, J. A. Lawrence, C. Sestacovschi, A. Tidd-Johnson, M. H. U. Rasool, V. K. Gadamidi, S. Ozair, K. Pandav, C. Cuevas-Lou, et al., "The impact of misinformation on the covid-19 pandemic," AIMS Public Health, vol. 9, no. 2, p. 262, 2022.
- [2] F. Ennab, M. S. Babar, A. R. Khan, R. J. Mittal, F. A. Nawaz, M. Y. Essar, and S. S. Fazel, "Implications of social media misinformation on covid-19 vaccine confidence among pregnant women in africa," *Clinical Epidemiology and Global Health*, vol. 14, p. 100981, 2022.
- [3] E. Team, "European digital media observatory." https://edmo.eu/.
- [4] I. Team, "International fact-checking network." https://www.poynter.org/ifcn/.
- [5] J. Ma, W. Gao, P. Mitra, S. Kwon, B. J. Jansen, K.-F. Wong, and M. Cha, "Detecting rumors from microblogs with recurrent neural networks," 2016.
- [6] F. Yu, Q. Liu, S. Wu, L. Wang, T. Tan, et al., "A convolutional approach for misinformation identification.," in *IJCAI*, pp. 3901–3907, 2017.
- [7] T. E. Trueman, A. Kumar, P. Narayanasamy, and J. Vidya, "Attention-based c-bilstm for fake news detection," *Applied Soft Computing*, vol. 110, p. 107600, 2021.
- [8] Y. Zhou, Y. Yang, Q. Ying, Z. Qian, and X. Zhang, "Multi-modal fake news detection on social media via multi-grained information fusion," in Proceedings of the 2023 ACM international conference on multimedia retrieval, pp. 343–352, 2023.
- [9] P. Li, X. Sun, H. Yu, Y. Tian, F. Yao, and G. Xu, "Entity-oriented multi-modal alignment and fusion network for fake news detection," *IEEE Transactions on Multimedia*, vol. 24, pp. 3455–3468, 2021.
- [10] J. Thorne, A. Vlachos, O. Cocarascu, C. Christodoulopoulos, and A. Mittal, "The fact extraction and verification (fever) shared task," arXiv preprint arXiv:1811.10971, 2018.
- [11] N. Kotonya and F. Toni, "Explainable automated fact-checking: A survey," arXiv: Computation and Language, arXiv: Computation and Language, Nov 2020.
- [12] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al., "Chain-of-thought prompting elicits reasoning in large language models," Advances in Neural Information Processing Systems, vol. 35, pp. 24824–24837, 2022.
- [13] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pretrain, prompt, and predict: A systematic survey of prompting methods in natural language processing," ACM Computing Surveys, vol. 55, no. 9, pp. 1–35, 2023.
- [14] J. Li, T. Tang, W. X. Zhao, J.-Y. Nie, and J.-R. Wen, "Pre-trained language models for text generation: A survey," ACM Computing Surveys, vol. 56, no. 9, pp. 1–39, 2024.
- [15] S. Yin, C. Fu, S. Zhao, K. Li, X. Sun, T. Xu, and E. Chen, "A survey on multimodal large language models," arXiv preprint arXiv:2306.13549, 2023.
- [16] X. Zhang and W. Gao, "Towards Ilm-based fact verification on news claims with a hierarchical step-by-step prompting method," arXiv preprint arXiv:2310.00305, 2023.

- [17] L. Pan, X. Wu, X. Lu, A. T. Luu, W. Y. Wang, M.-Y. Kan, and P. Nakov, "Fact-checking complex claims with program-guided reasoning," arXiv preprint arXiv:2305.12744, 2023.
- [18] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, et al., "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," arXiv preprint arXiv:2311.05232, 2023.
- [19] P. Nakov, D. Corney, M. Hasanain, F. Alam, T. Elsayed, A. Barrón-Cedeño, P. Papotti, S. Shaar, and G. Martino, "Automated fact-checking for assisting human fact-checkers," arXiv: Artificial Intelligence, arXiv: Artificial Intelligence, Mar 2021.
- [20] X. Hu, Z. Guo, J. Chen, L. Wen, and P. S. Yu, "Mr2: A benchmark for multimodal retrieval-augmented rumor detection in social media," in *Proceedings of the 46th international ACM SIGIR conference on* research and development in information retrieval, pp. 2901–2912, 2023.
- [21] B. M. Yao, A. Shah, L. Sun, J.-H. Cho, and L. Huang, "End-to-end multimodal fact-checking and explanation generation: A challenging dataset and models," in *Proceedings of the 46th International ACM SI-GIR Conference on Research and Development in Information Retrieval*, pp. 2733–2743, 2023.
- [22] S. Singhal, A. Kabra, M. Sharma, R. R. Shah, T. Chakraborty, and P. Kumaraguru, "Spotfake+: A multimodal framework for fake news detection via transfer learning (student abstract)," in *Proceedings of the* AAAI conference on artificial intelligence, vol. 34, pp. 13915–13916, 2020
- [23] W.-W. Du, H.-W. Wu, W.-Y. Wang, and W.-C. Peng, "Team triple-check at factify 2: Parameter-efficient large foundation models with feature representations for multi-modal fact verification," arXiv preprint arXiv:2302.07740, 2023.
- [24] S. Abdelnabi, R. Hasan, and M. Fritz, "Open-domain, content-based, multi-modal fact-checking of out-of-context images via online resources," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14940–14949, 2022.
- [25] F. Alam, S. Cresci, T. Chakraborty, F. Silvestri, D. Dimitrov, G. D. S. Martino, S. Shaar, H. Firooz, and P. Nakov, "A survey on multimodal disinformation detection," arXiv preprint arXiv:2103.12541, 2021.
- [26] Y. Zhu, Q. Sheng, J. Cao, S. Li, D. Wang, and F. Zhuang, "Generalizing to the future: Mitigating entity bias in fake news detection," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2120–2125, 2022.
- [27] L. Xiao, Q. Zhang, C. Shi, S. Wang, U. Naseem, and L. Hu, "Msynfd: Multi-hop syntax aware fake news detection," in *Proceedings of the ACM on Web Conference 2024*, pp. 4128–4137, 2024.
- [28] K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu, "defend: Explainable fake news detection," in *Proceedings of the 25th ACM SIGKDD international* conference on knowledge discovery & data mining, pp. 395–405, 2019.
- [29] L. Wu, Y. Rao, Y. Lan, L. Sun, and Z. Qi, "Unified dual-view cognitive model for interpretable claim verification," arXiv preprint arXiv:2105.09567, 2021.
- [30] Y. Chen, D. Li, P. Zhang, J. Sui, Q. Lv, L. Tun, and L. Shang, "Cross-modal ambiguity learning for multimodal fake news detection," in *Proceedings of the ACM web conference* 2022, pp. 2897–2905, 2022.
- [31] L. Wang, C. Zhang, H. Xu, Y. Xu, X. Xu, and S. Wang, "Cross-modal contrastive learning for multimodal fake news detection," in *Proceedings* of the 31st ACM international conference on multimedia, pp. 5696–5704, 2023
- [32] M. Akhtar, M. Schlichtkrull, Z. Guo, O. Cocarascu, E. Simperl, and A. Vlachos, "Multimodal automated fact-checking: A survey," arXiv preprint arXiv:2305.13507, 2023.
- [33] T. Alhindi, S. Petridis, and S. Muresan, "Where is your evidence: Improving fact-checking by justification modeling," in *Proceedings of the first workshop on fact extraction and verification (FEVER)*, pp. 85–90, 2018.
- [34] K. Jiang, R. Pradeep, and J. Lin, "Exploring listwise evidence reasoning with t5 for fact verification," in Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pp. 402–410, 2021.
- [35] L. Wu, L. Wang, and Y. Zhao, "Unified evidence enhancement inference framework for fake news detection," in *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 6541–6549.

- [36] J. Ma, W. Gao, S. Joty, and K.-F. Wong, "Sentence-level evidence embedding for claim verification with hierarchical attention networks," Association for Computational Linguistics, 2019.
- [37] K. Popat, S. Mukherjee, A. Yates, and G. Weikum, "Declare: Debunking fake news and false claims using evidence-aware deep learning," arXiv preprint arXiv:1809.06416, 2018.
- [38] P. Atanasova, "Generating fact checking explanations," in Accountable and Explainable Methods for Complex Reasoning over Text, pp. 83–103, Springer, 2024.
- [39] L. Hu, S. Wei, Z. Zhao, and B. Wu, "Deep learning for fake news detection: A comprehensive survey," AI open, vol. 3, pp. 133–155, 2022.
- [40] Z. Guo, M. Schlichtkrull, and A. Vlachos, "A survey on automated fact-checking," *Transactions of the Association for Computational Lin*guistics, vol. 10, pp. 178–206, 2022.
- [41] B. Hu, Q. Sheng, J. Cao, Y. Shi, Y. Li, D. Wang, and P. Qi, "Bad actor, good advisor: Exploring the role of large language models in fake news detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 22105–22113, 2024.
- [42] K. M. Caramancion, "News verifiers showdown: a comparative performance evaluation of chatgpt 3.5, chatgpt 4.0, bing ai, and bard in news fact-checking," in 2023 IEEE Future Networks World Forum (FNWF), pp. 1–6, IEEE, 2023.
- [43] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "Fever: a large-scale dataset for fact extraction and verification," arXiv preprint arXiv:1803.05355, 2018.
- [44] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al., "Gpt-4o system card," arXiv preprint arXiv:2410.21276, 2024.
- [45] G. Team, P. Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, G. Tanzer, D. Vincent, Z. Pan, S. Wang, et al., "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," arXiv preprint arXiv:2403.05530, 2024.
- [46] G. Team, "Google programmable search engine." https://developers. google.com/custom-search.
- [47] T. Team, "Tineye reverse image search." https://tineye.com/.
- [48] Y. Team, "Yandex image search." https://yandex.com/images.
- [49] G. Team, "Google vision." https://cloud.google.com/vision/?hl=en.
- [50] S. Team, "Selenium automates browsers." https://www.selenium.dev/.
- [51] N. Team, "Newspaper3k: Article scraping & curation." https:// newspaper.readthedocs.io/en/latest/.
- [52] M. P. Polak and D. Morgan, "Extracting accurate materials data from research papers with conversational language models and prompt engineering," *Nature Communications*, vol. 15, no. 1, p. 1569, 2024.
- [53] F. news detection policy Of HKBU FACT CHECK Team. https:// factcheck.hkbu.edu.hk/home/en/fact-check/our-process/.
- [54] Wikipedia. https://en.wikipedia.org/wiki/List_of_fake_news_websites.
- [55] M. Bias. https://mediabiasfactcheck.com.
- [56] A. Olteanu, S. Peshterliev, X. Liu, and K. Aberer, "Web credibility: Features exploration and credibility prediction," in Advances in Information Retrieval: 35th European Conference on IR Research, ECIR 2013, Moscow, Russia, March 24-27, 2013. Proceedings 35, pp. 557– 568, Springer, 2013.
- [57] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping languageimage pre-training with frozen image encoders and large language models," in *International conference on machine learning*, pp. 19730– 19742, PMLR, 2023.
- [58] X. Wei, X. Cui, N. Cheng, X. Wang, X. Zhang, S. Huang, P. Xie, J. Xu, Y. Chen, M. Zhang, et al., "Chatie: Zero-shot information extraction via chatting with chatgpt," arXiv preprint arXiv:2302.10205, 2024.
- [59] S. Team, "Snopes fact checks." https://www.snopes.com/fact-check/.
- [60] P. Team, "Politifact fact check." https://www.politifact.com/.
- [61] J. Chi, U. Karn, H. Zhan, E. Smith, J. Rando, Y. Zhang, K. Plawiak, Z. D. Coudert, K. Upasani, and M. Pasupuleti, "Llama guard 3 vision: Safe-guarding human-ai image understanding conversations," arXiv preprint arXiv:2411.10414, 2024.
- [62] J. Bai, S. Bai, S. Yang, S. Wang, S. Tan, P. Wang, J. Lin, C. Zhou, and J. Zhou, "Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond," 2023.
- [63] O. Team, "Openai platform web search." https://platform.openai.com/ docs/guides/tools-web-search?api-mode=chat.
- [64] G. Team, "Ai for developers, grounding with google search." https://ai.google.dev/gemini-api/docs/grounding?lang=python.

- [65] N. Hassan, C. Li, and M. Tremayne, "Detecting check-worthy factual claims in presidential debates," in *Proceedings of the 24th acm in*ternational on conference on information and knowledge management, pp. 1835–1838, 2015.
- [66] B. Adair, C. Li, J. Yang, and C. Yu, "Progress toward "the holy grail": The continued quest to automate fact-checking," in *Computation+ Journalism Symposium*, (September), 2017.

APPENDIX

A. Parameters Justification

To explore the optimized hyperparameter α in Section V-A3, we set a group of α with different values (0.4, 0.5, 0.6) and evaluate the performance of Holmes in RQ2 in the MMDV dataset with four metrics, respectively. The Holmes is deployed with GPT-40, Gemini-1.5-flash, Llama 3.2-vision, and Qwen-vision. The results are shown in Table XI, and the best performance of each model among different values of α is in bold. In general, these four models achieve the best performance, setting α as 0.5, indicating a balance between the relevance and integrity of the evidence. The results suggest that relevance and integrity are equally significant when filtering out high-quality evidence for fact-checking.

TABLE XI: Performance of Holmes deployed with different LLMs under different value of α .

Model	α	Accuracy	Precision	Recall	F1
	0.4	88.4%	89.1%	87.9%	88.5%
GPT-4o	0.5	90.2%	89.9%	90.0%	89.8%
	0.6	87.1%	88.2%	85.9%	87.1%
	0.4	84.7%	86.3%	84.0%	85.1%
Gemini-1.5-flash	0.5	87.0%	87.8%	86.7%	86.8%
	0.6	84.0%	86.0%	82.7%	84.3%
	0.4	71.2%	73.7%	42.7%	54.1%
Llama 3.2-vsion	0.5	73.4%	70.8%	56.0%	48.1%
	0.6	70.9%	73.1%	41.2%	52.5%
	0.4	68.2%	52.4%	41.9%	46.4%
Qwen-vision	0.5	71.1%	39.6%	43.4%	41.6%
	0.6	68.4%	47.9%	38.6%	42.7%

B. Verification Cost and Efficiency of Holmes

We consider the cost of utilizing Holmes for disinformation verification. The cost of Holmes is incurred by invoking commercial APIs (LLMs API and Google API). We utilize T_{total} and $Cost_{total}$ to denote the total execution time and the total invocation cost, respectively. For open-source LLMs, we only compute the elapsed time(T_{total}). The computing formulas are shown in Equation 5 and Equation 6.

$$T_{total} = T_{retrieve} + T_{summarize} + T_{verify}$$
 (5)

$$Cost_{total} = Cost_{retrieve} + Cost_{summarize} + Cost_{verify}$$
(6)

The total execution time (T_{total}) and the total invocation cost $(Cost_{total})$ of the verification process consist mainly of the following three parts. Note that the framework initializes two threads to execute text direct search and image reverse

TABLE XII: Time cost (s) per disinformation verification. The first column from the left indicates the three stages in this process.

	GPT-40	Gemini-1.5-flash	Llamma 3.2-Vision	Qwen-VL
Retrieve	0.1	0.1	0.1	0.1
Summary	10.3	9.7	80.2	82.4
Verify	4.2	2.0	20.1	23.8
Total	14.6	11.8	100.4	106.3

search in parallel, rather than sequentially, to save as much time as possible.

- 1) Retrieve evidence ($T_{retrieve}$, $Cost_{retrieve}$): Time and cost of Invoking the Google text direct search engine and the image reverse search engine to search for information related to target claim from the Internet.
- 2) Summarize main content ($T_{summarize}$, $Cost_{summarize}$): Time and cost of Invoking the LLM API to summarize the main content of the original web page.
- 3) Verify claims $(T_{verify}, Cost_{verify})$: Time and cost of invoking the LLM API to verify the claim using the retrieved evidence.

The time cost of Holmes with different LLMs is shown in Table XII. Overall, Commercial LLMs (13.2s on average) are faster than open-source LLMs (103.4s). Gemini-1.5-flash has the shortest elapsed time of 11.8s. We compute the fees according to the billing rules according to the vendors' portal websites ⁵. GPT-40 incurs a small fee of 0.055 USD for each disinformation verification, whereas Gemini-1.5-flash provides free access. The most expensive and most time-consuming step during real-time fact check is the summary, as it requires processing massive amounts of text and image data when summarizing the main content from the original web pages. Compared to professional fact-check agents, which require several hours or days to verify a piece of disinformation on average [65], [66], our method is extremely cost-effective, which greatly reduces elapsed time.

C. LLM Prompt Designs in Holmes

This section provides the full set of prompt templates used in Holmes. These prompts were designed to instruct LLMs in completing different subtasks during the disinformation verification.

The following template is guiding Holmes to summarize the main content of a webpage in Section V-A2.

Suppose you are a professional fact-checker.

Please summarize the provided article by identifying the people (who), the event (what), the location (where), the time (when), the reason (why), the background of the event, the impact of the event, and the follow-up event. Ensure the summary remains concise and clear.

The following template is guiding Holmes to initialize a disinformation verification task in Section V-B.

Suppose you are a professional fact-checker. I will give you a claim to verify. The following is the claim. {text} denotes the text part of the claim. {image} denotes the image part of the claim.

Text: {text}
Image: {image}

Before I provide you with evidence to verify this claim, do nothing but memorize it.

The following template is guiding Holmes to upload evidence in Section V-B.

The following list is the evidence related to the claim. You need to remember it and do nothing until the next instruction.

Text evidence: {text_evidence_list}

The following template is guiding Holmes to verify the claim in Section V-B.

Verify the claim based on the evidence that I provided to you. The verdict sets of the claim and the verification principle is shown below.

True verdict set: {true_verdict_set}. False verdict set: {false_verdict_set}.

(1) If your verification result is in the true verdict set, the claim is true. (2) If your verification result is in the false verdict set, the claim is false.

Next, give the justification for the verdict result. Output your complete answers in the format of the following template.

{output_format}

The following template is guiding Holmes to output verification results in an explicit format in Section V-B.

Verdict: True/False.

Evidence:

- 1. The evidence {place_holder} supports/refutes thep-lace holder of the claim.
- 2. The evidence {place_holder} supports/refutes thep-lace_holder of the claim.

3.

Summary: Use a concise sentence to summarize including your prediction and reason.

⁵https://openai.com/api/pricing/